Novelty only

, INTERNATIONAL APPLI[redacted]ON PUBLISHED UNDER THE PATENT COO[redacted]TION TREATY (PCT)

(9) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 December 2002 (05.12.2002)

PCT

(10) International Publication Number
WO 02/097677 A2

(51) International Patent Classification[7]: G06F 17/30

(21) International Application Number: PCT/US02/16928

(22) International Filing Date: 31 May 2002 (31.05.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/294,705 31 May 2001 (31.05.2001) US
09/938,322 23 August 2001 (23.08.2001) US

(71) Applicant: **INTEL CORPORATION** [US/US]; 2625 Walsh Avenue, Santa Clara, CA 95051 (US).

(72) Inventors: **HE, Liang**: A-18, JiuShi Western Suburban Garden, 168 LaoHuQingPing Road, Shangai (CN). **ZHU, Xiaogang**: Room 502, Block 7, 959 Hong Qiao Road, Chang Ning District, Shangai 200051 (CN). **YEUNG, Chung, Yun**: Flat B, 9/F, Block 21, Parc Oasis, Grandeur Road, Kowloon, Hong Kong (CN). **ZHU, Jing, Yong**: Suite 502, Building 8, Lane 535, LongHuaXi Road, Shanghai 200232 (CN). **YANG, Kefeng**: Dorm N°110, Building n°15, East China Normal University, Shanghai (CN). **XIE, Chuanquan**: Room 201, N° 1, Graduate Student Bldg., 1953 Hua Shan Road, Shanghai Jiao Tong University, Shanghai 200030 (CN).

(74) Agents: **STEFFEY, Charles, E.** et al.; Schwegman, Lundberg, Woessner & Kluth, P.O. Box 2938, Minneapolis, MN 55402 (US).

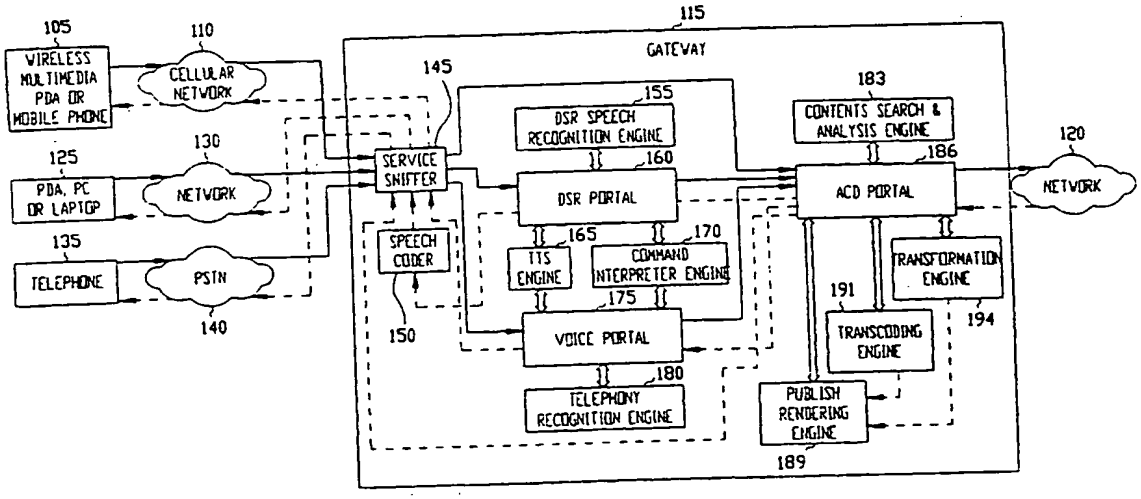
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: INFORMATION RETRIEVAL CENTER



(57) Abstract: A method, apparatus, and gateway allows a variety of client devices to access information from current information providers, and the information is adapted to accommodate the differing formats of the client services. Speech processing, text-to-speech (TTS), speech feature transmission on heterogeneous network, quality of service (QoS) support for real-time applications, transcoding, transformation, publish rendering, multimedia content analysis, and speech coding may be supported.

WO 02/097677 A2

INFORMATION RETRIEVAL CENTER

FIELD

This invention relates generally to computers and more particularly to information retrieval using remote devices.

COPYRIGHT NOTICE/PERMISSION

BACKGROUND

As the Internet has evolved, users have begun to access information on remote servers using small devices, such as handheld devices, cellular phones, laptop computers, and PDAs (Personal Digital Assistants). These small client devices have smaller displays, less powerful processors, less memory, and different user interfaces than traditional laptop or desktop computers.

Some of these devices suffer from problems when attempting to interface with current servers, which were designed to accommodate traditional desktop computers. First, some devices have awkward and difficult input techniques, such as selecting letters on a screen using a stylus or finger. Second, since current information providers have designed their data on the servers to have a format convenient for and compatible with traditional desktop computers, there can be a format mismatch between the data and the device attempting to receive the data and communicate it to the user. Third, the format of the data is often inconvenient for the users of small devices. Finally, small devices with a slow processor and small memory cannot easily handle the large amounts of data from current servers.

Thus, there is a need for a solution that will enable a variety of devices to take advantage of the data created by information providers.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 depicts a block diagram of an embodiment of the invention.

Fig. 2 depicts a block diagram of a stack of function blocks, according to an embodiment of the invention.

Fig. 3 depicts a block diagram of an exemplary electronic device according to an embodiment of the invention.

Fig. 4 depicts a flowchart of processing when the client device is a wireless or wire handset and speech is used for information retrieval.

Fig. 5 depicts a flowchart of processing when the client device is a wireless or wire handset and a keyboard or pen is used for information retrieval.

Fig. 6 depicts a flowchart of processing when the client device is a desktop or laptop computer and speech is used for information retrieval.

Fig. 7 depicts a flowchart of processing when the client device is a desktop or laptop computer and a keyboard or pen is used for information retrieval.

Fig. 8 depicts a flowchart of processing when the client device is a telephone and speech is used for information retrieval.

DETAILED DESCRIPTION

In the following detailed description of exemplary embodiments of the invention, reference is made to the accompanying drawings (where like numbers represent like elements), which form a part hereof, and in which is shown by way of illustration specific exemplary embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention, but other embodiments may be

utilized and logical, mechanical, electrical, and other changes may be made without departing from the scope of the present invention. The following detailed description is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims.

In the following description, numerous specific details are set forth to provide a thorough understanding of the invention. However, it is understood that the invention may be practiced without these specific details. In other instances, well-known circuits, structures and techniques have not been shown in detail in order not to obscure the invention.

Fig. 1 depicts a block diagram of an embodiment of the invention. Wireless Multimedia, a PDA, and a mobile phone 105 are shown communicatively coupled to cellular network 110, which is communicatively coupled to gateway 115. Gateway 115 is shown communicatively coupled to network 120, such as the Internet or an Intranet. PDA, PC (Personal Computer), or laptop 125 is shown communicatively coupled to Network 130, which is communicatively coupled to gateway 115. In an embodiment, network 130 may be the Internet, but in other embodiments any suitable network may be used. Telephone 135 is shown communicatively coupled to PSTN (Public Switched Telephone Network) 140, which is communicatively coupled to gateway 115. Gateway 115 includes service sniffer 145, speech coder 150, DSR (Distributed Speech Recognition) speech recognition engine 155 , DSR portal 160, TTS (Text To Speech) engine 165, command interpreter engine 170, voice portal 175, telephony recognition engine 180, contents search and analysis engine 183, ACD (Adaptive Content Delivery) portal 186, publish rendering engine 189, transcoding engine 191, and transformation engine 194. In an embodiment, gateway 115 may be a functional unit that interconnects two networks with different network architectures. In an embodiment gateway 115 has the hardware configuration described below with respect to Fig. 3.

Fig. 2 depicts a block diagram of stacks of function blocks, according to an embodiment of the invention. Client 205, transmission medium 240, gateway 250, and network monitor 299 are shown.

Client device 205 represents functions in any of client devices 105, 125, and 135, as previously described above with reference to Fig. 1. Referring again to Fig. 2, client device 205 includes speech decoder 207, transformation decoder 209, contents display 211, DSR feature extractor 213, QoS (Quality of Service) module 215, client applications 216, transport controller 218, DSR feature compressor 219, and contents ornamentor 221. In other embodiments, client device 205 may include only selected functions.

Speech decoder 207 may decompress the compressed audio from gateway 115. Transformation decoder 209 may decode the transformed multimedia contents from gateway 115.

Contents Ornamentor 221 may ornament black-and-white multimedia contents to a higher-resolution multiple-scale gray format or higher-resolution pseudo color. Contents Ornamentor 221 also may minus adapt the multi-media contents for display on the screen size of client 205.

Contents display 211 adapts the received contents for proper display according to the physical properties of the client device. Examples of physical properties are screen size, screen resolution, and color depth. But, in other embodiments, any physical properties of the client device may be used. Contents display 211 processes the output of speech decoder 207, transformation decoder 209, and contents ornamentor 221 to finally display the contents at client 205. In an embodiment, the properties of the client device may be determined using the User Agent protocol, which provides a description of client device capabilities. In another embodiment, the CC/PP (Composite Capabilities/Preference Profiles) protocol may be used, which provides a description of client device capabilities and user preferences. In other embodiments, any suitable technique for determining the capabilities of the client device may be used. When client 205 initiates a session by negotiating with gateway 115, the type of client information (whether User Agent format, CC/PP format, or some other format) may be sent to inform gateway 115 about the client's capabilities.

DSR feature compressor 219 encodes the speech feature that may be sent to gateway 115. DSR feature extractor 213 extracts the speech feature from the user's speech. In an embodiment DSR feature extractor 213 uses MFCC (Mel-Frequency Cepstral Coefficients) to extract the speech feature. In an embodiment, the HMM (Hidden Markov Model) may be used for speech recognition. QoS Module 215 may be an interface to network monitor 299. QoS Module 215 accepts QoS information from network monitor 299 and gives feedback information to gateway 115. Transport Controller 218 receives QoS information from QoS Module 215 and then transfers it to gateway 115, or vice versa. Transport controller 218 also starts transmission for heterogenous network bearers.

Network Monitor 299 on the client side monitors the information stream from gateway 115 to client 205, measures QoS parameters for this information stream, and reports this QoS information to client applications 216.

Components stack in transmission medium 240 includes DSR transmission protocols 242, IP (Internet Protocol) 244, and Wireless/Wireline Bearer 246. DSR transmission protocol 242 may be an application layer protocol that may be above IP 244. With QoS module 215 and transport controller 218, DSR transmission protocol 242 transmits compressed speech and retrieves information efficiently and robustly. IP 244 may be a protocol that governs the breakup of data messages into packets, the routing of the packets from sender to destination, and the reassembly of the packets into the original data messages at the destination. Wireless/Wireline bearer 246 may be a transport protocol that transmits audio packets across a transmission medium. Examples of appropriate wireless/wireline bearers include GPRS (General Packet Radio Service), EDGE (Enhanced Data Rate for GSM and TDMA/136 Evolution), 3G (Third Generation Mobile Communication System), and the Internet. In other embodiments, any appropriate wireless/wireline protocol may be used.

Gateway stack 250 represents a stack of functions in gateway 115. Gateway stack 250 includes service sniffer 145, speech coder 150, DSR speech

recognition engine 155, DSR portal 160, TTS engine 165, command interpreter engine 170, voice portal 175, telephony recognition engine 180, contents search and analysis engine 183, ACD portal 186, publish rendering engine 189, transcoding engine 191, and transformation engine 194.

When client 205 initiates a session by negotiating with gateway 115, the type of client information (whether User Agent format, CC/PP format, or some other format) may be sent to inform gateway 115 about the client's capabilities. This client information may be used by ACD portal 186 to adapt the contents presentation via publish rendering engine 189, transformation engine 194, and transcoding Engine 191.

Publish rendering engine 189 renders a particular format, in an embodiment HTML (Hypertext Markup Language) or WML (Wireless Markup Language), for client 205. In other embodiments, any appropriate format may be rendered. Examples of the functions of publish rendering engine 189 include converting one display page into multiple pages or converting one display line into multiple lines, including fractions of lines. But, any rendering function may be used, and the invention is not so limited.

Transformation engine 194 converts a data format used in the contents retrieved from network 120 into a format supported by client device 205. For example, some clients may not be able to display GIF (Graphics Interchange Format) images because of software or hardware constraints, so transformation engine 194 converts the GIF images into a format that the client does support, such as BMP (raster graphics stored in bitmap format), JPEG (Joint Photographic Experts Group), or black-and-white images. But, the invention is not limited to these particular formats.

Transcoding Engine 191 converts the original content retrieved from network 120 within multiple fidelities and modalities and selects the proper content version for particular clients 205. Typical examples of fidelity transcoding include summarizing, including image compression, text summarization, and video key-frame extraction. In other embodiments, any

appropriate fidelity transcoding may be used. Modality transcoding may be translation, such as conversion from text to audio, or video to images. In other embodiments, any appropriate translation may be used.

Contents search and analysis engine 183 receives keywords from DSR Portal 160, voice portal 175, or directly from service sniffer 145. According to the given keywords, contents search and analysis engine 183 searches network 120 for appropriate contents, analyzes the contents, extracts the contents most relevant to the keywords, and then returns the results to DSR portal 160, voice portal 175, or directly to service sniffer 145.

TTS engine 165 translates text in the retrieved contents into audio speech. Speech coder 150 compresses audio to accommodate the bandwidth of transmission medium 240. Command interpreter engine 170 detects and interprets the purpose for user's inputs especially for the methodology of voice inputs. Telephony recognition engine 180 performs speech recognition for voice input from client 135. Speech recognition engine performs speech recognition for DSR portal 160.

Service sniffer 145 distinguishes between different inputs from clients and directs them to the appropriate services within gateway 115. The different inputs from clients may be classified into three basic types: traditional telephone services (which are directed to voice portal 175), forthcoming DSR services (which are directed to DSR portal 160), and routine IP based services (which are directed to ACD portal 186). Service sniffer 145 acts as an interface between transmission medium 240 and gateway 115.

Voice portal 175 controls the input and output of telephony recognition engine 180. Voice portal 175 sends speech data to telephony recognition engine 180 and retrieves the recognition result back. Either TTS engine 165 or command interpreter engine 170 processes the recognition result and either returns the recognition result to client 205 or uses the recognition result to request more services, for example by using the recognition result to search network 120.

DSR portal 160 performs analogous functions as voice portal 175, but operates in a DSR environment and processes speech feature data.

ACD portal 186 receives information retrieving commands and searches for the information in the Internet using contents search and analysis engine 183. The results of the search are processed by transformation engine 194 or transcoding engine 191, after which the results are returned to client 205 in the particular format supplied by publish rendering engine 189.

DSR synchronization 254 lies in the DSR receiver side and synchronizes received DSR bit-streams. DSR feature decoding 258 decodes the compressed feature data from clients 205.

QoS Daemon 256 receives QoS requesting information from client 205, adjusts the QoS parameters of the client according to practical network conditions, and then sends these modified QoS parameters back to client 205.

Transport Controller 252 supervises QoS information of the network and other network states and then reports the state information to QoS Daemon 256.

Fig. 3 shows a block diagram illustrating an exemplary electronic device 300 according to an embodiment of the invention. The exemplary electronic device 300 includes processor 305, storage device 310, input device 320, output device 327, and network adapter 330, all communicatively coupled via bus 315. An unillustrated network may also be coupled to bus 315 through network adapter 330.

Processor 305 represents a central processing unit of any type of architecture, such as a CISC (Complex Instruction Set Computing), RISC (Reduced Instruction Set Computing), VLIW (Very Long Instruction Word), or a hybrid architecture, although any appropriate processor may be used. Processor 305 executes instructions and includes that portion of electronic device 300 that controls the operation of the entire electronic device. Although not depicted in Fig. 1, processor 305 typically includes a control unit that organizes data and program storage in computer memory and transfers data and

other information between the various parts of the electronic device. Processor 305 receives input data from input device 320 and network adapter 330, reads and stores code and data in storage device 310, and optionally presents output data to a user via output device 327. Processor 305 also sends and receives packets of information across a network using network adapter 330.

Although electronic device 300 is shown to contain only a single processor and a single system bus, the present invention applies equally to computers that have multiple processors and to computers that have multiple buses that each performs different functions in different ways.

Storage device 310 represents one or more mechanisms for storing data. For example, storage device 310 may include read only memory (ROM), random access memory (RAM), magnetic disk storage mediums, optical storage mediums, flash memory devices, and/or other machine-readable mediums. Although only one storage device 310 is shown, multiple storage devices and multiple types of storage devices may be present. Further, although electronic device 300 is drawn to contain storage device 310, the storage device may be distributed across other electronic devices. Storage device 310 contains code 340, which has instructions capable of being executed on processor 305 to carry out the functions of the present invention. In another embodiment, the functions of the present invention may be carried out via hardware in lieu of a processor-based system. Of course, storage device 310 may also contain additional software and data (not shown), which is not necessary to understanding the invention.

Bus 315 represents one or more busses (e.g., PCI (Peripheral Component Interconnect), ISA (Industry Standard Architecture), X-Bus, EISA (Extended Industry Standard Architecture), or any other appropriate bus) and bridges (also termed bus controllers).

Input device 320 is that part of electronic device 300 that accepts input from a user. In an embodiment, input device 320 may be a keyboard, but in other embodiments, input device 320 may be a pointing device, mouse,

trackball, keypad, touchpad, touch screen, pointing stick, microphone, or any other appropriate input device.

Output device 327 communicates information to the user of electronic device 300. Output device 327 may be a cathode-ray tube (CRT) based video display well known in the art of computer hardware. But, in other embodiments output device 327 may be replaced with a liquid crystal display (LCD) based or gas, plasma-based, flat-panel display. In other embodiments, any appropriate display device is used. In still other embodiments, a speaker that produces audio output may be used. In other embodiments, output device 327 might not be present.

Network adapter 330 facilitates communication between electronic device 300 and an unillustrated network. Network adapter 330 provides a user of electronic device 300 with a means of electronically communicating information, such as packets, with a remote computer or a network logical-storage device. In addition, in another embodiment, network adapter 330 supports distributed processing, which enables electronic device 300 to share a task with other devices linked to the network. Although network adapter 330 is shown as part of electronic device 300, in another embodiment they may be packaged separately.

Electronic device 300 may be implemented using any suitable hardware and/or software, such as a personal computer available from a number of vendors. Portable computers, laptop computers, mainframe computers, handheld devices, PDAs, telephones, cellular telephones, and network computers or Internet appliances are examples of other possible configurations. Electronic device 300 is an example implementation of devices 105, 125, 135, and/or 115. The hardware and software depicted in Fig. 3 may vary for specific applications and may include more or fewer elements than those depicted. For example, other peripheral devices such as pointing devices, speech recognition devices, audio adapters, or chip programming devices, such as EPROM (Erasable Programmable Read-Only Memory) programming devices may be

used in addition to or in place of the hardware already depicted. Thus, an embodiment of the invention may apply to any hardware configuration.

As will be described in detail below, aspects of an embodiment pertain to specific apparatus and method elements implementable on computers. In another embodiment, the invention may be implemented as a computer program product for use with a computer system. The programs defining the functions of this embodiment may be delivered to a computer via a variety of signal-bearing media, which include, but are not limited to:

(1) information permanently stored on non-rewriteable storage media (e.g., read only memory devices within a computer such as CD-ROM readable by a CD-ROM drive;

(2) alterable information stored on rewriteable storage media (e.g., a hard disk drive or diskette); or

(3) information conveyed to a computer by a communications media, such as through a computer or telephone network accessed via network adapter 330, including wireless communications.

Such signal-bearing media, when carrying processor-readable instructions that direct the functions of the present invention, represent embodiments of the present invention.

Fig. 4 depicts a flowchart of processing when the client device is a wireless or wire handset and speech is used for information retrieval. Control begins at block 400. Control then continues to block 405 where client 105 performs speech feature extraction. Control then continues to block 410 where client 105 performs feature compression. Control then continues to block 415 where client 105 transmits the compressed data to gateway 115. Control then continues to block 420 where DSR portal 160 performs speech recognition on the compressed data. Control then continues to block 425 where command interpreter engine 170 determines the user request from the output of DSR portal 160 and sends the user request to ACD portal 186. Control then continues to

block 430 where ACD portal 186 retrieves the contents associated with the user request from network 120. Control then continues to block 435 where ACD portal 186 adapts retrieved contents based on network bandwidth considerations and the client devices's resources. Control then continues to block 440 where ACD portal 186 sends the adapted contents to DSR portal 160.

Control then continues to block 445 where DSR portal 160 sends some of the adapted contents directly to client 105. DSR portal 160 sends those contents directly to client 105 that link to the URL (Uniform Resource Locator) of a video, image, or audio database. A URL may be an address in a network. DSR portal 160 also sends some of the adapted contents to TTS 165, which converts the contents to speech and then sends the speech to speech coder 150, which compresses the speech and sends it to client 105. DSR portal 160 chooses adapted contents to send to TTS 165 based on information in text adapted contents that indicates the contents are text intended to be published to users in audio speech format. In an embodiment, this indication may be supplied by a prompt tag, although in other embodiments any appropriate indication may be used.

Control then continues to block 450 where client 105 decodes the contents and communicates it to the user. Control then continues to block 499 where the function returns.

Fig. 5 depicts a flowchart of processing when the client device is a wireless or wire handset and a keyboard or pen is used for information retrieval. Control begins at block 500. Control then continues to block 505 where client 105 gets a URL from the keyboard or pen input. Client 105 sends the URL to ACD portal 186 in gateway 115. Control then continues to block 510 where ACD portal 186 retrieves the contents associated with the URL from network 120. Control then continues to block 515 where ACD portal 186 adapts the retrieved contents based on network bandwidth and the resources of client device 105. Control then continues to block 520 where ACD portal 186 sends back the adapted contents to DSR portal 160.

Control then continues to block 525 where DSR portal 160 sends some of the adapted contents directly to client 105. DSR portal 160 also sends some of the adapted contents to TTS 165, which converts the contents to speech and then sends the speech to speech coder 150, which compresses the speech and sends it to client 105. Control then continues to block 530 where client 105 decodes the encoded contents and communicates them to the user. Control then continues to block 599 where the function returns.

Fig. 6 depicts a flowchart of processing when the client device is a desktop or laptop computer and speech is used for information retrieval. Control begins at block 600. Control then continues to block 605 where client 125 performs speech-feature extraction. Control then continues to block 610 where client 125 performs feature compression. Control then continues to block 615 where client 125 transmits the compressed data to gateway 115. Control then continues to block 620 where DSR portal 160 performs speech recognition on the compressed data. Control then continues to block 625 where command interpreter engine 170 determines the URL associated with the output of DSR portal 160 and sends the URL to ACD portal 186. Control then continues to block 630 where ACD portal 186 retrieves the contents associated with the URL. Control then continues to block 640 where ACD portal 186 sends the contents to DSR portal 160. Control then continues to block 645 where DSR portal 160 sends some of the contents directly to client 125. DSR portal 160 also sends some of the contents to TTS 165, which converts the contents to speech and then sends the speech to speech coder 150, which compresses the speech and sends it to client 125. Control then continues to block 650 where client 125 decodes the contents and communicates it to the user. Control then continues to block 699 where the function returns.

Fig. 7 depicts a flowchart of processing when the client device is a desktop or laptop computer and a keyboard or pen is used for information retrieval. Control begins at block 700. Control then continues to block 705 where client 125 obtains a URL from the keyboard or pen input. Client 125 sends the URL to ACD portal 186 in gateway 115. Control then continues to

block 710 where ACD portal 186 retrieves the contents associated with the URL from network 120. Control then continues to block 720 where ACD portal 186 sends back the contents to DSR portal 160. Control then continues to block 725 where DSR portal 160 sends some of the contents directly to client 125. DSR portal 160 also sends some of the contents to TTS 165, which converts the contents to speech and then sends the speech to speech coder 150, which compresses the speech and sends it to client 125. Control then continues to block 730 where client 125 decodes the encoded contents and communicates them to the user. Control then continues to block 799 where the function returns.

Fig. 8 depicts a flowchart of processing when the client device is a telephone and speech is used for information retrieval. Control begins at block 800. Control then continues to block 805 where client 135 sends user audio to gateway 115 via PSTN 140. Control then continues to block 810 where voice portal 175 performs speech recognition. Control then continues to block 815 where command interpreter engine 170 determines the URL based on the output of voice portal 175. Control then continues to block 820 where ACD portal 186 retrieves contents from network 120 based on the URL. Control then continues to block 840 where TTS engine 165 translates the retrieved contents to audio. Control then continues to block 845 where TTS engine 165 transfers the audio to client 135. Control then continues to block 850 where client 135 plays the audio to the user. Control then continues to block 899 where the function returns.

WHAT IS CLAIMED IS:

1. A gateway, comprising:

a command interpreter engine to detect keywords in speech input;

a search and analysis engine to search a network for contents based on the keywords; and

a transformation engine to convert a data format used in the contents retrieved from the network into a format supported by a client device.

2. The gateway of claim 1, wherein the transformation engine is to convert an image from one format into another format.

3. The gateway of claim 1, further comprising:

a service sniffer to distinguish between different inputs from different clients and to direct the different inputs to appropriate services within the gateway.

4. The gateway of claim 3, wherein the service sniffer is to direct telephone services to a voice portal.

5. The gateway of claim 3, wherein the service sniffer is to direct DSR (distributed speech recognition) services to a DSR portal.

6. The gateway of claim 3, further comprising:

a quality of service daemon to receive quality of service requesting information from the client.

7. The gateway of claim 6, wherein the quality of service daemon is further to adjust quality of service parameters of the client device according to network conditions and then to send the adjusted quality of service parameters to the client device.

8. The gateway of claim 1, further comprising:

a text-to-speech engine to translate text in the contents into audio speech.

9. The gateway of claim 1, further comprising:

a speech coder to compress audio to accommodate bandwidth of a transmission medium between the client device and the gateway.

10. The gateway of claim 1, further comprising:

a publish rendering engine to convert a display page into multiple pages.

11. The gateway of claim 1, further comprising:

a publish rendering engine to convert a display line into multiple lines.

12. A method, comprising:

extracting a feature from user input;

translating the feature into a request;

retrieving contents from a network based on the request; and

adapting the contents to a client.

13. The method of claim 12, wherein the adapting further comprises converting text to audio speech.

14. The method of claim 12, wherein the adapting further comprises adapting the contents to a screen size of the client.

15. The method of claim 12, wherein the adapting further comprises adapting the contents to a screen resolution of the client.

16. The method of claim 12, wherein the adapting further comprises adapting the contents to a color depth of the client.

17. The method of claim 12, wherein the adapting further comprises converting a display page into multiple pages.

18. The method of claim 12, wherein the adapting further comprises converting a display line into multiple lines.

19. The method of claim 12, wherein the user input comprises an address of the contents.

20. The method of claim 19, wherein the address is a uniform resource locator.

21. The method of claim 12, wherein the feature further comprises at least one keyword in the user input.

22. A program product comprising signal-bearing media, wherein the signal-bearing media comprises instructions, wherein the instructions when read and executed comprise:

extracting a feature from user speech;

translating the feature into a request;

retrieving contents from a network based on the request; and

adapting the contents for transmission to a telephone.

23. The program product of claim 22, wherein the feature comprises a keyword to be searched.

24. The program product of claim 22, wherein the adapting further comprises:

translating text in the contents into audio speech.

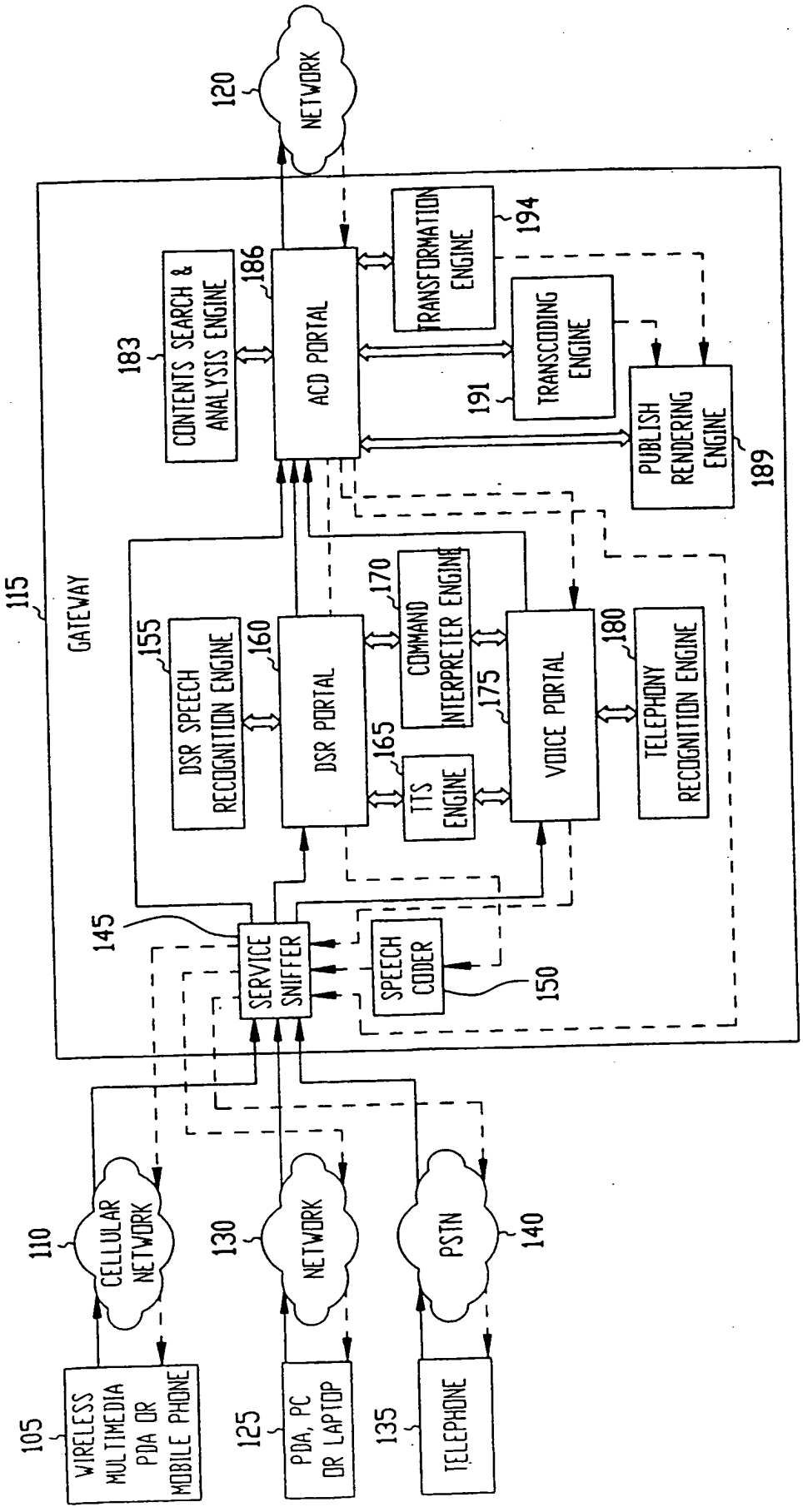

105
WIRELESS MULTIMEDIA PDA OR MOBILE PHONE
110
CELLULAR NETWORK
125
PDA, PC OR LAPTOP
130
NETWORK
135
TELEPHONE
140
PSTN
115
GATEWAY
145
SERVICE SNIFFER
150
SPEECH CODER
155
DSR SPEECH RECOGNITION ENGINE
160
DSR PORTAL
165
TTS ENGINE
170
COMMAND INTERPRETER ENGINE
175
VOICE PORTAL
180
TELEPHONY RECOGNITION ENGINE
183
CONTENTS SEARCH & ANALYSIS ENGINE
186
ACD PORTAL
189
PUBLISH RENDERING ENGINE
191
TRANSCODING ENGINE
194
TRANSFORMATION ENGINE
120
NETWORK

FIG. 1

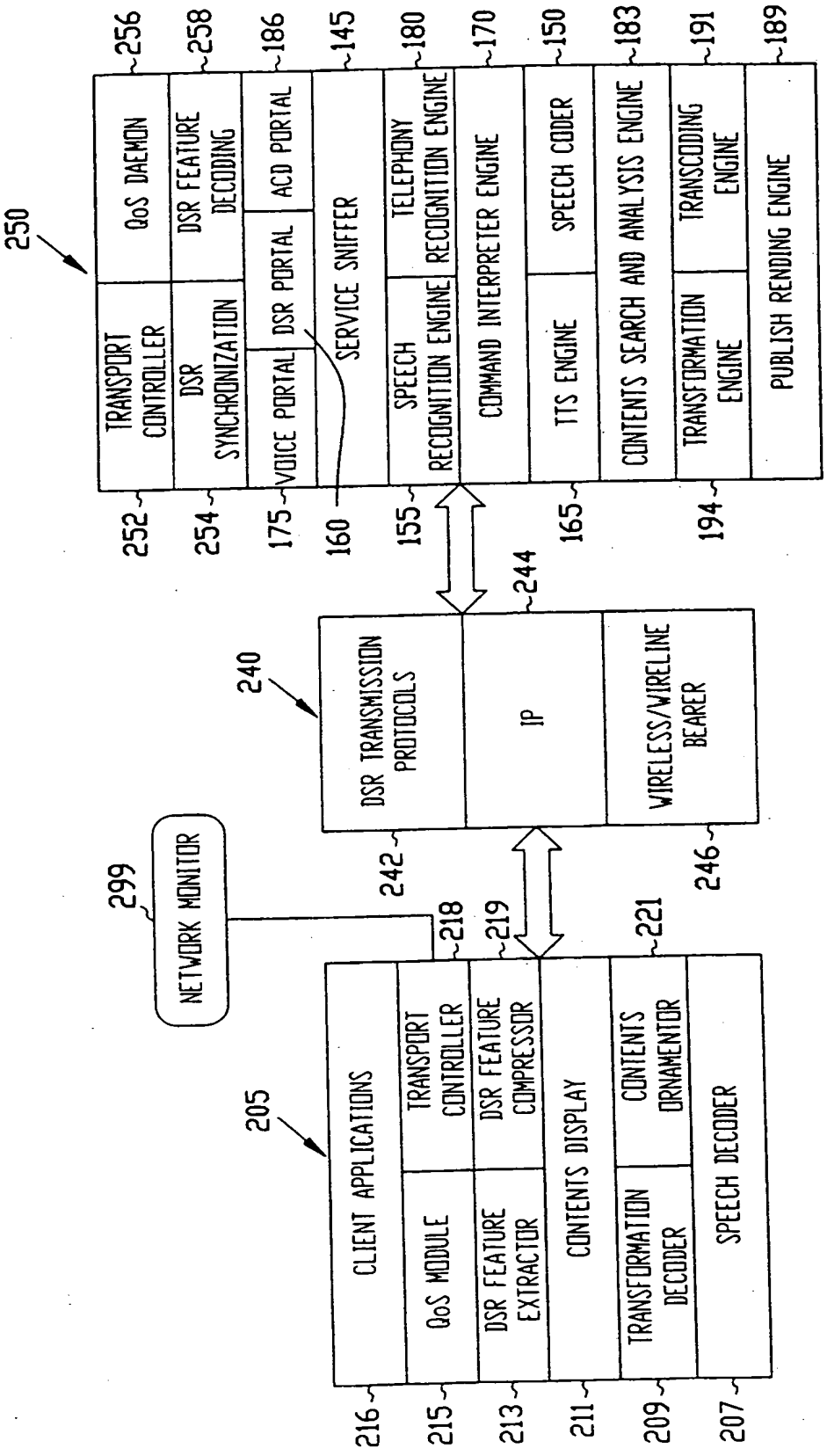

250
TRANSPORT CONTROLLER
QoS DAEMON
DSR SYNCHRONIZATION
DSR FEATURE DECODING
VOICE PORTAL
DSR PORTAL
ACD PORTAL
SERVICE SNIFFER
SPEECH RECOGNITION ENGINE
TELEPHONY RECOGNITION ENGINE
COMMAND INTERPRETER ENGINE
TTS ENGINE
SPEECH CODER
CONTENTS SEARCH AND ANALYSIS ENGINE
TRANSFORMATION ENGINE
TRANSCODING ENGINE
PUBLISH RENDING ENGINE
256
258
186
145
180
170
150
183
191
189
252
254
175
160
155
165
194
240
DSR TRANSMISSION PROTOCOLS
IP
WIRELESS/WIRELINE BEARER
244
242
246
299
NETWORK MONITOR
218
219
221
205
CLIENT APPLICATIONS
QoS MODULE
TRANSPORT CONTROLLER
DSR FEATURE EXTRACTOR
DSR FEATURE COMPRESSOR
CONTENTS DISPLAY
TRANSFORMATION DECODER
CONTENTS ORNAMENTOR
SPEECH DECODER
216
215
213
211
209
207

FIG. 2

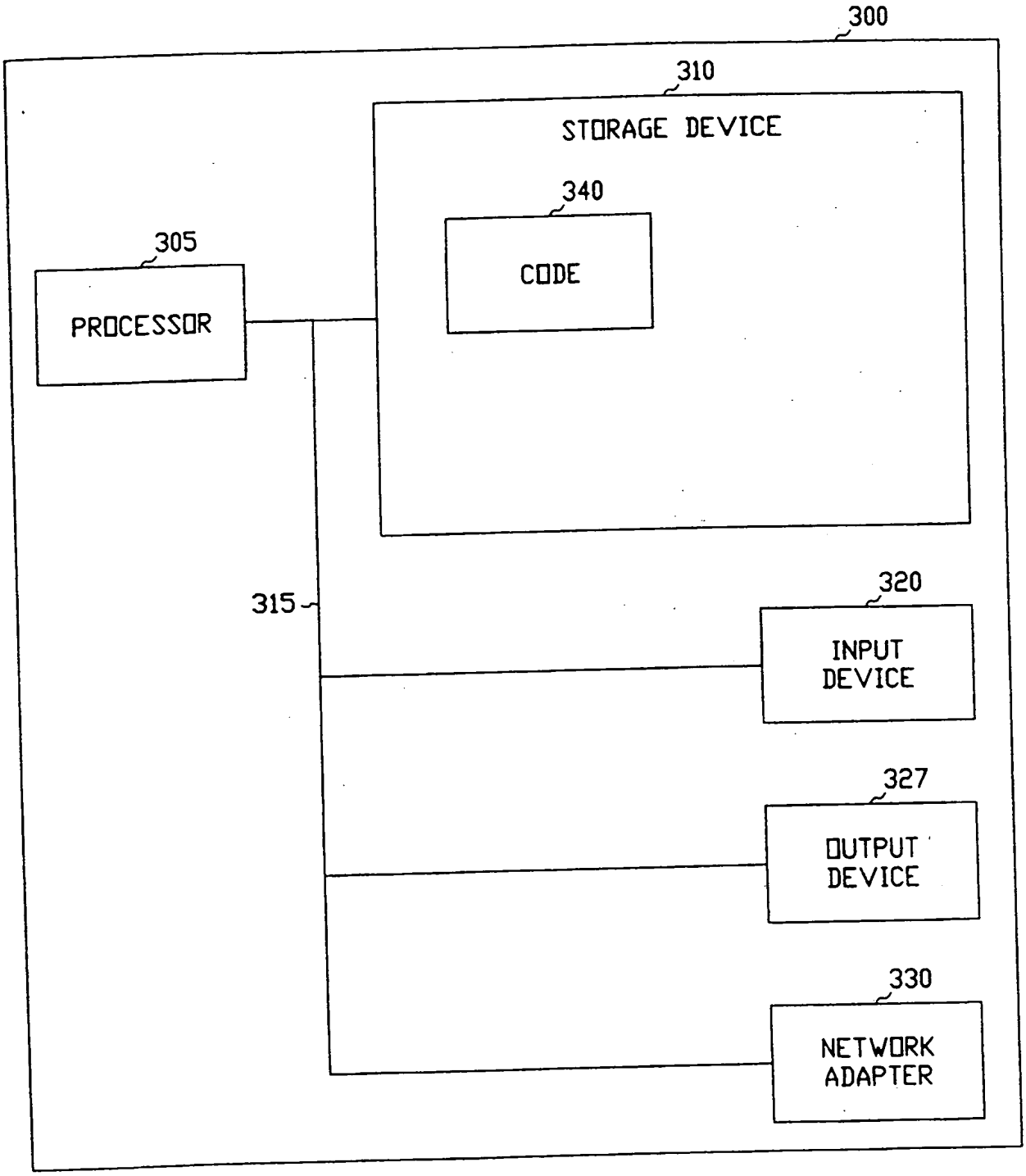

300
310
STORAGE DEVICE
340
CODE
305
PROCESSOR
315
320
INPUT DEVICE
327
OUTPUT DEVICE
330
NETWORK ADAPTER

FIG. 3

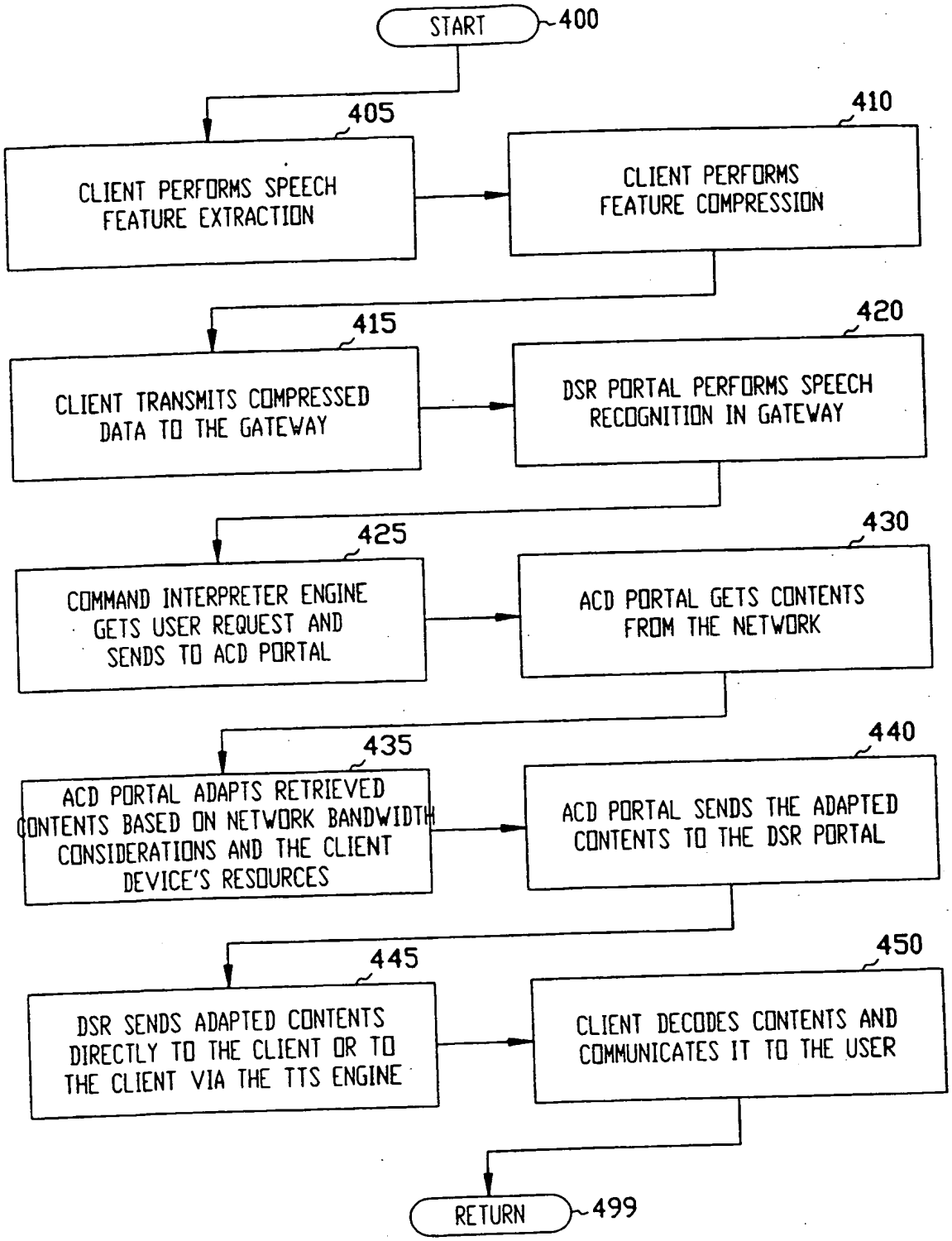

START
400
405
CLIENT PERFORMS SPEECH FEATURE EXTRACTION
410
CLIENT PERFORMS FEATURE COMPRESSION
415
CLIENT TRANSMITS COMPRESSED DATA TO THE GATEWAY
420
DSR PORTAL PERFORMS SPEECH RECOGNITION IN GATEWAY
425
COMMAND INTERPRETER ENGINE GETS USER REQUEST AND SENDS TO ACD PORTAL
430
ACD PORTAL GETS CONTENTS FROM THE NETWORK
435
ACD PORTAL ADAPTS RETRIEVED CONTENTS BASED ON NETWORK BANDWIDTH CONSIDERATIONS AND THE CLIENT DEVICE'S RESOURCES
440
ACD PORTAL SENDS THE ADAPTED CONTENTS TO THE DSR PORTAL
445
DSR SENDS ADAPTED CONTENTS DIRECTLY TO THE CLIENT OR TO THE CLIENT VIA THE TTS ENGINE
450
CLIENT DECODES CONTENTS AND COMMUNICATES IT TO THE USER
RETURN
499

FIG. 4

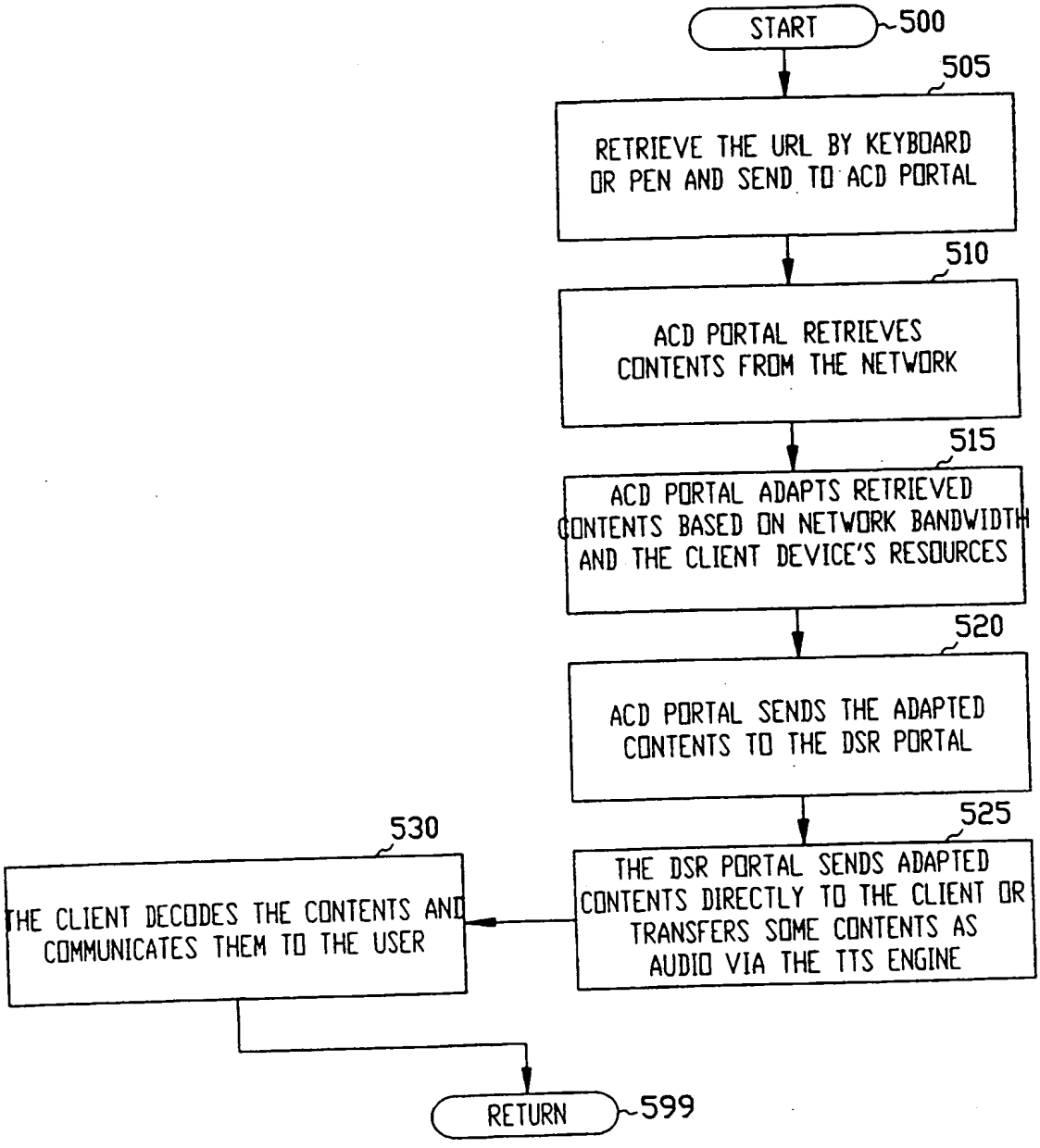

START
500
505
RETRIEVE THE URL BY KEYBOARD OR PEN AND SEND TO ACD PORTAL
510
ACD PORTAL RETRIEVES CONTENTS FROM THE NETWORK
515
ACD PORTAL ADAPTS RETRIEVED CONTENTS BASED ON NETWORK BANDWIDTH AND THE CLIENT DEVICE'S RESOURCES
520
ACD PORTAL SENDS THE ADAPTED CONTENTS TO THE DSR PORTAL
525
THE DSR PORTAL SENDS ADAPTED CONTENTS DIRECTLY TO THE CLIENT OR TRANSFERS SOME CONTENTS AS AUDIO VIA THE TTS ENGINE
530
THE CLIENT DECODES THE CONTENTS AND COMMUNICATES THEM TO THE USER
RETURN
599

FIG. 5

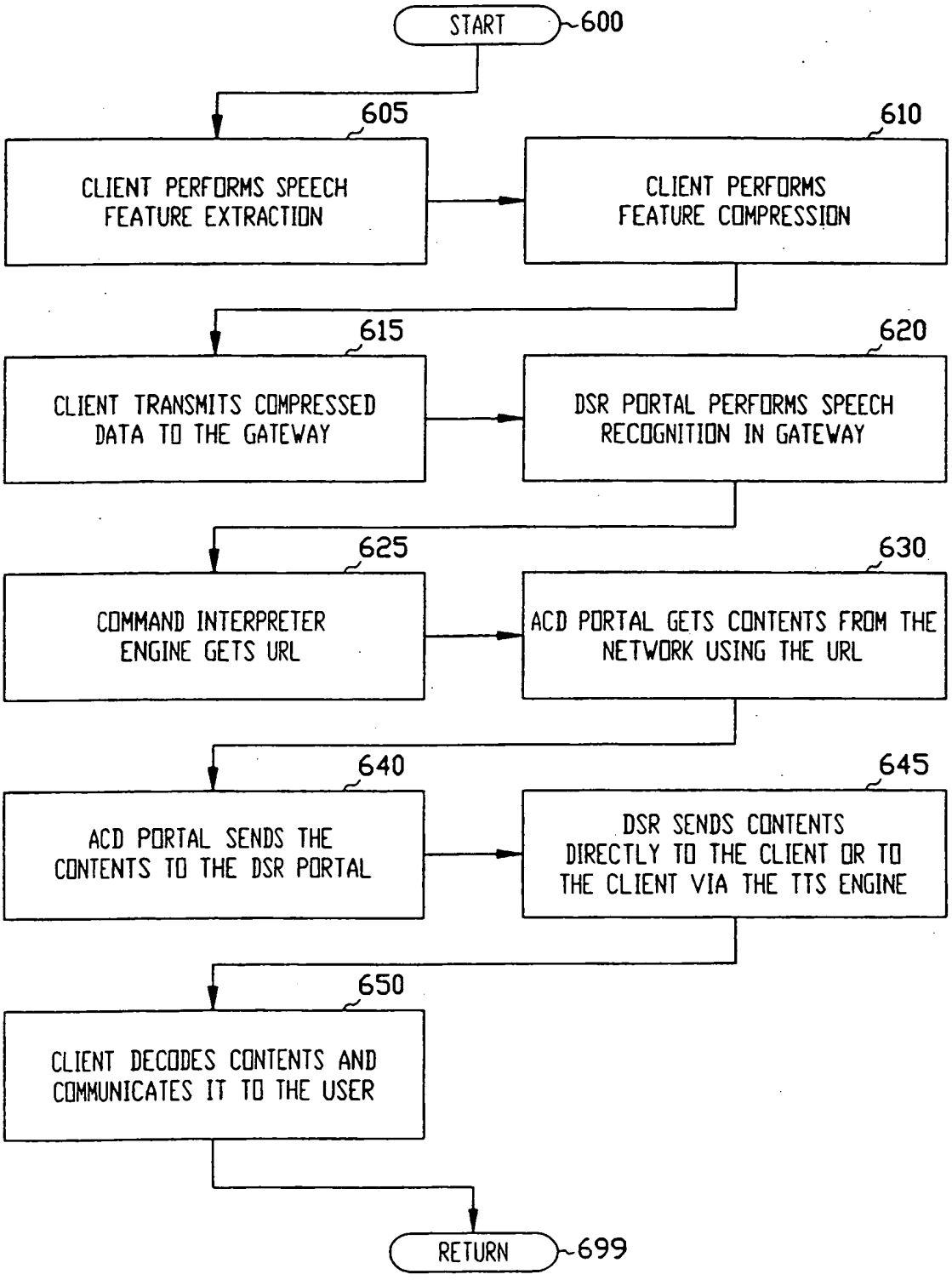

START
600
605
CLIENT PERFORMS SPEECH FEATURE EXTRACTION
610
CLIENT PERFORMS FEATURE COMPRESSION
615
CLIENT TRANSMITS COMPRESSED DATA TO THE GATEWAY
620
DSR PORTAL PERFORMS SPEECH RECOGNITION IN GATEWAY
625
COMMAND INTERPRETER ENGINE GETS URL
630
ACD PORTAL GETS CONTENTS FROM THE NETWORK USING THE URL
640
ACD PORTAL SENDS THE CONTENTS TO THE DSR PORTAL
645
DSR SENDS CONTENTS DIRECTLY TO THE CLIENT OR TO THE CLIENT VIA THE TTS ENGINE
650
CLIENT DECODES CONTENTS AND COMMUNICATES IT TO THE USER
RETURN
699

FIG. 6

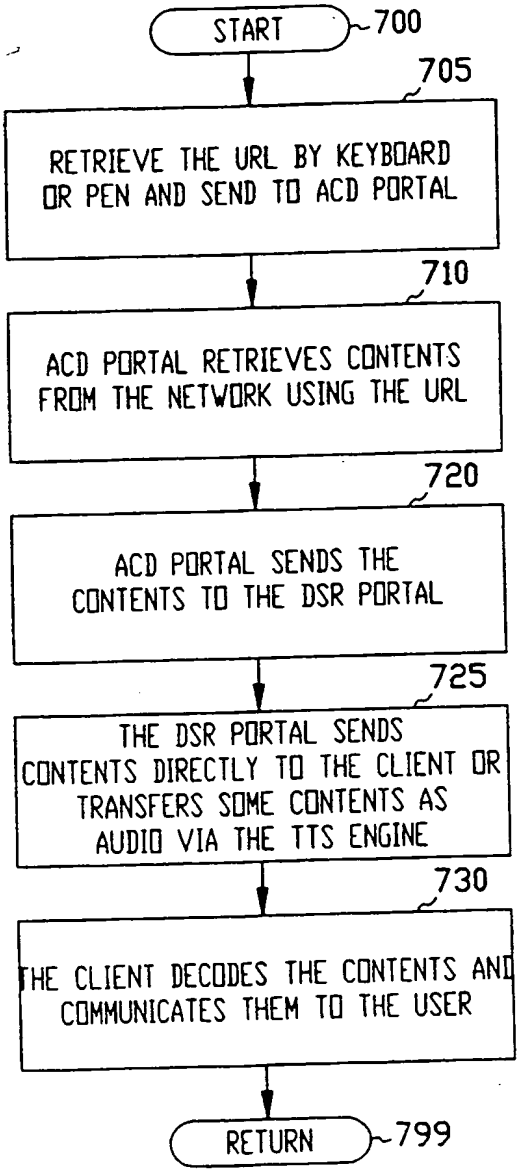

START
700
705
RETRIEVE THE URL BY KEYBOARD OR PEN AND SEND TO ACD PORTAL
710
ACD PORTAL RETRIEVES CONTENTS FROM THE NETWORK USING THE URL
720
ACD PORTAL SENDS THE CONTENTS TO THE DSR PORTAL
725
THE DSR PORTAL SENDS CONTENTS DIRECTLY TO THE CLIENT OR TRANSFERS SOME CONTENTS AS AUDIO VIA THE TTS ENGINE
730
THE CLIENT DECODES THE CONTENTS AND COMMUNICATES THEM TO THE USER
RETURN
799

FIG. 7

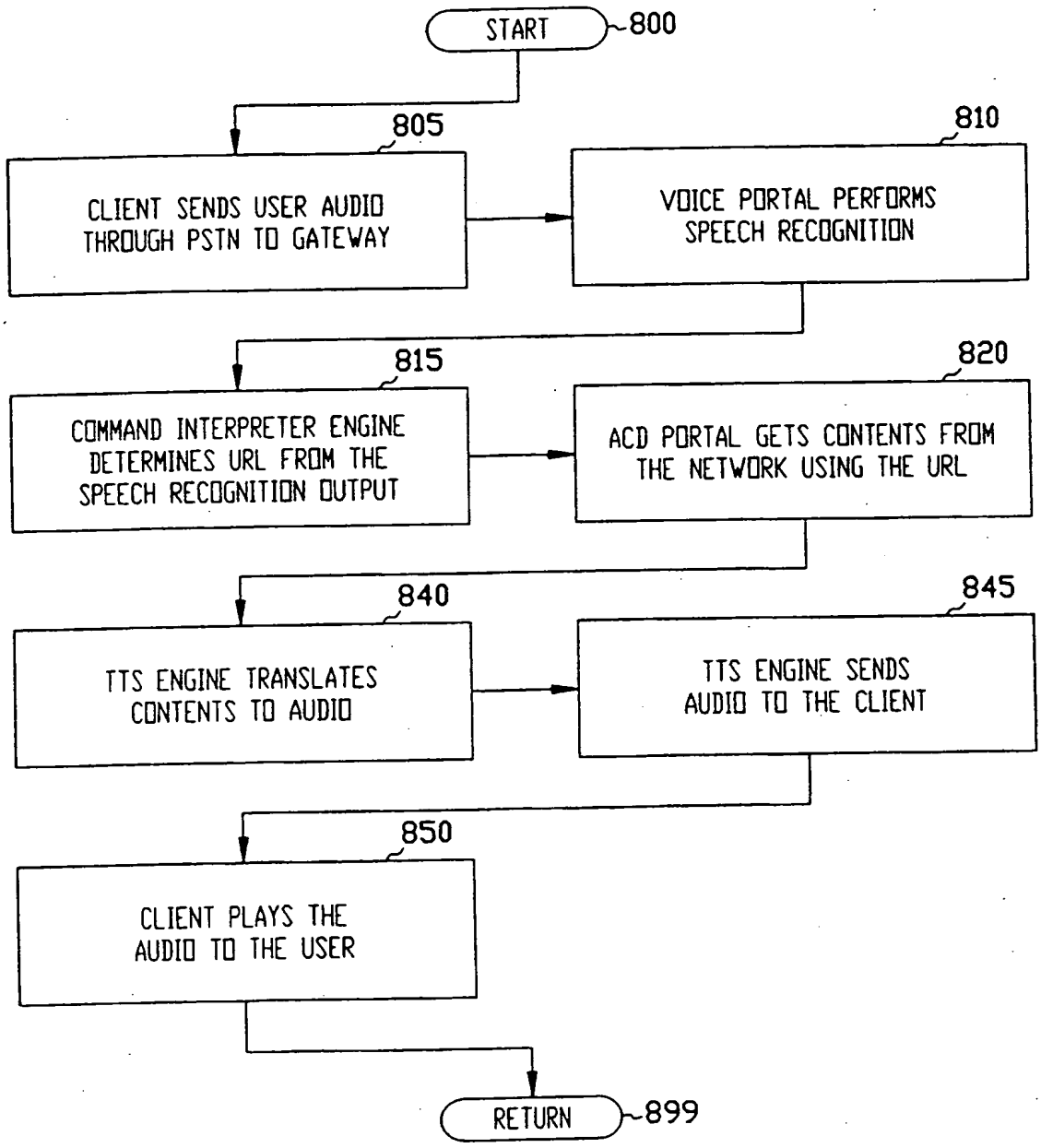

START
800
805
CLIENT SENDS USER AUDIO THROUGH PSTN TO GATEWAY
810
VOICE PORTAL PERFORMS SPEECH RECOGNITION
815
COMMAND INTERPRETER ENGINE DETERMINES URL FROM THE SPEECH RECOGNITION OUTPUT
820
ACD PORTAL GETS CONTENTS FROM THE NETWORK USING THE URL
840
TTS ENGINE TRANSLATES CONTENTS TO AUDIO
845
TTS ENGINE SENDS AUDIO TO THE CLIENT
850
CLIENT PLAYS THE AUDIO TO THE USER
RETURN
899

FIG. 8